1Д

УДК 681.142.2/518.3

П.Н. Денисенко
Кировоградский национальный технический университет, г. Кировоград, Украина
pnden_osvita@yahoo.com

Алгоритм решения краевых задач в системах компьютерной алгебры по τ-методу Ланцоша

В статье построен алгоритм τ-метода Ланцоша для решения многоточечных линейных краевых задач для линейных дифференциальных уравнений порядка k с многочленными коэффициентами. По этому алгоритму в компьютерных системах символьного преобразования вычисляют многочлен порядка n . Доказана эквивалентность этого алгоритма и алгоритма применения a-метода Дзядыка. Результаты исследования a-метода доказывают существование решения исходной задачи по алгоритму, сходимость последовательности таких решений (с ростом параметра n алгоритма) к точному решению краевой задачи и точные и конструктивные априорные и апостериорные оценки погрешности в пространствах $C_{[a,b]}$, $C^k_{[a,b]}$ для достаточно широкого класса уравнений и краевых условий.

Введение

Задача. *Построить алгоритм решения многоточечной линейной краевой задачи для ЛДУМК* (task) *на отрезке* $[a, b]$ *в системах компьютерной алгебры (СКА). По этому алгоритму вычисляют решение, оптимальное для символьных преобразований в СКА.*

Многоточечная линейная краевая задача для ЛДУМК – это система, состоящая из линейного дифференциального уравнения с многочленными коэффициентами [1]

$$D[y] + G= 0, \ D[y] = A\, y^{(k)} + \ldots + C\, y \qquad (1)$$

и k линейных краевых условий в точках $d_i = d_i$, $i = 1,\ldots, k$, отрезка $[a, b]$

$$Cond(y) = \{D_i[y] = G_i\,|_{x=d_i}\},\ d_i = d_i \in [a, b],\ i = 1,\ldots, k\,\}; \qquad (2)$$

где $D_i[y]$ – линейный дифференциальный оператор с постоянными коэффициентами

$$D_i[y] = A_i\, y^{(k_i)} + \ldots + C_i\, y$$

имеет порядок $k_i = k_i = ord_equ(D_i[y]) < k = ord_equ(D[y])$. Точки задания условий (2) являются не особыми точками ЛДУМК (1) – $A(d_i) \neq 0,\ i = 1,\ldots, k$.

Актуальность задачи. Maple, Mathematica, Mathcad, Matlab, APS и другие компьютерные системы стали естественной средой математического моделирования. Эти системы символьно преобразуют композиции специальных математических функций (СМФ) и решают дифференциальные уравнения – являются СКА. Краевые задачи (1), (2) являются классическим аппаратом математического моделирования [1].

Возможности СКА. СКА не имеют процедур для решения отдельных классов краевых задач. Система Maple вычисляет аналитическое решение обыкновенных дифференциальных уравнений в следующем виде:
– композиция СМФ (она существует в очень редких случаях),
– частная сумма ряда Тейлора решения задачи Коши порядка n-. Порядок этого многочлена, как правило, меньше 10. Ряд Тейлора является аппаратом аппроксимации функций, не оптимальным по точности. Следовательно, частная сумма ряда Тейлора решения задачи Коши для ЛДУМК (1) (вычисленная системой Maple) часто не удовлетворяет основное требование математического моделирования – **точность**.

Алгоритм [2] решает класс краевых задач для ЛДУМК (1), (2). Для краевых условий этого класса задач существует многочлен порядка $k - 1$

$$y_{k-1} = solve(Cond(y_{k-1} \in P_{k-1})) \quad (3)$$

и оператор

$$K[u] = solve(y^{(k)} = u,\ \{D_i[y]\ _{x=d_i} = 0,\ i = 1,\ldots,k\}),\ P_p \to P_{p+k} \quad (4)$$

линейного преобразования функции u в первообразную этой функции порядка k

$$K[u] + y_{k-1}\text{: } u = y^{(k)} \to y,\ C_{[a,b]} \to C^k_{[a,b]}\,.$$

Эта первообразная удовлетворяет краевые условия (2). Многочлен $y_{k-1} \in P_{k-1}$ уравнений $Cond(y_{k-1} \in P_{k-1})$ имеет порядок $k - 1$, символьные коэффициенты и является элементом общего вида *множества алгебраических многочленов порядка k – 1.*

По алгоритму [2] вычисляют алгебраический многочлен порядка n

$$y_n = c_0 + c_1 x + \ldots + c_n x^n = algorithm\ (task\ (1), (2), [a,b], n). \quad (5)$$

Коэффициент оптимальности алгоритма [2] в пространстве $C^k_{[a,b]}$ ограничен

$$C_n(algorithm\ ,\ task\ (1), (2),\ C^k_{[a,b]}) = \|\ y - y_n\ \|_{C^\wedge k_[a,b]} / E_n(y,\ C^k_{[a,b]}) = O(1) \quad (6)$$

В коэффициенте оптимальности (6) эталоном для нормы погрешности алгоритма на краевой задаче (1), (2) в пространстве $C^k_{[a,b]} = C^\wedge k_[a,b]$

$$\|\ y - y_n\ \|_{C^\wedge k_[a,b]} = \max\ \{\|\ y - y_n\ \|_{C[a,b]}\ ,\ \|\ (y - y_n)'\ \|_{C[a,b]\ ,\ldots,}\ \|\ (y - y_n)^{(k)}\|_{C[a,b]}\}$$

является величина наилучшего приближения точного решения y исходной задачи *task* алгебраическими многочленами порядка n в этом пространстве

$$E_n(y,\ C^k_{[a,b]}) = \inf_{\{c_0,\ \ldots\ ,\ c_n\ \}}\ \|\ y(x) - (c_0 + \ \ldots\ + c_n\ x^n)\ \|_{C^\wedge k_[a,b]}.$$

Недостаток алгоритма [2]. СКА не преобразуют условия (2) в оператор $K[u]$ (4).

Метод решения задачи. *Построить алгоритм τ-метода Ланцоша* [1], *эквивалентный алгоритму* [2].

1 Алгоритм 1

Вход: $D[y] + G = 0$ (1) , $Cond(y)$ (2), $[a,b]$, n.
Выход: Многочлен (5).
Преобразования:

1. Вычислить аппроксимацию ЛДУМК (1)

$$D[y_n \in P_n] + G + (E_{m,p}(z(x)) \in H_{m\backslash p}) = 0. \quad (7)$$

2. Вычислить аппроксимацию краевых условий (2)

$$Cond(y_n \in P_n) = \{D_i[\ y_n \in P_n]\ |_{x=d_i} + G_i = 0,\ i = 1,\ldots,k\}. \quad (8)$$

3. Вычислить решение системы уравнений (7), (8) – многочлены

$$\{y_n,\ E_{m,p}(z(x))\} = solve(D[y_n \in P_n\ ] + G + (E_{m,p}(z(x)) \in H_{m\backslash p}\ ) = 0,\ Cond(y_n \in P_n). \quad (9)$$

В уравнениях (7) и (8) многочлен $y_n \in P_n$ имеет вид (5) и символьные коэффициенты. Поэтому он является элементом общего вида множества P_n – многочленов порядка n.

Многочлен $E_{m,p}(z(x)) \in H_{m\backslash p}$ уравнения (7) является дополнительным многочленом а-метода Дзядыка [3] с параметром p решения линейных интегральных уравнений Вольтерра с многочленными коэффициентами. Этот многочлен имеет параметры:

$$m = \deg(D[\ y_n \in P_n\ ] + G), \quad (10)$$

$$p = n - k, \qquad (11)$$

где $k = ord_equ(D[y])$ – порядок старшей производной оператора $D[y]$, вид

$$(E_{m,p}(z(x)) \in H_{m\backslash p}) = subs(x = z(x), E_{m,p}), \qquad (12)$$

где

$$z(x) = 2\,(x - a) / (b - a) - 1, \qquad (13)$$

$$E_{m,p} = \tau_1\, cheb(p + 1, x) + \ldots + \tau_{m-p}\, cheb(m, x) \qquad (14)$$

– дополнительный многочлен а-метода Дзядыка на отрезке [–1, 1]. Базис этого многочлена является ортогональным базисом пространства Гильберта $L_2(a,b;\rho)$

$$cheb(0, z(x)), cheb(1, z(x)), \ldots, cheb(i, x) = \cos(i \arccos(x)). \qquad (15)$$

Многочлен $E_{m,p}(z(x)) \in H_{m\backslash p}$ имеет символьные коэффициенты $(\tau_1,\ldots, \tau_{m-p}) = (\tau_1(n),\ldots, \tau_{m-p}(n))$. Поэтому он является многочленом общего вида пространства

$H_{m\backslash p}$ – *линейной оболочки элементов базиса* (15) *с индексом* $i = p + 1,\ldots, m$.

2 Алгоритм 1 и классические методы

Модельная краевая задача метода сплайнов [4, с. 63]

$$y'' = 100\, y,\; y(0) = 1,\; y(1) = 1 \qquad (16)$$

имеет единственное решение

$$y = K[u] + 1 = \mathrm{ch}(10\, x - 5) / \mathrm{ch}(5),\;\; u = y'' = 100\, y, \qquad (17)$$

где

$$K[u] = \int_0^x (1 - x)\, t\, u(t)\, dt + \int_x^1 (1 - t)\, x\, u(t)\, dt\,.$$

Функция y (17) – целая. Отличные от нуля коэффициенты Фурье – Чебышева функции y (17) на отрезке [0,1] только четные

$$\{a_{2\,i}(y, [0, 1])\}_{i=1}^{11} = \{0.47, 0.14, 0.021, 2\;10^{-3}, 1.2\;10^{-4}, 5.4\;10^{-6}, 1.7\;10^{-7}, 4.3\;10^{-9}, 8.5\;10^{-11}, 1.4\;10^{-12}, 1.8\;10^{-14}\}. \qquad (18)$$

С ростом параметра i они регулярно убывают $|\, a_{2\,i}(y\,, [0,1])\,| = o(q^i)$, $q \approx 0.1$. Следовательно, справедливо тождество

$$E_n(\,y, C_{[0,1]}\,) = (1 + o(1))\,|\, a_{2\,[n/2]+2}(y, [0, 1])\,|. \qquad (19)$$

Вычислительный эксперимент с алгоритмом 1. Решение краевой задачи (16) по алгоритму 1 на отрезке [0,1] является четным многочленом и справедливо тождество

$$y_n = algorithm_1(task\ (16), [0, 1], n\,) = y_{2[n/2]}.$$

Этот многочлен тождественен решению задачи (16) по алгоритму [2]. Норма погрешности решения краевой задачи (16) по алгоритму 1 в пространстве $C_{[0,1]}$ убывает с ростом параметра n алгоритма

$$\{\|\, y - y_{2i}\,\|_{C[0,1]} = \|\, y - y_{2i+1}\,\|_{C[0,1]}\,\}_{i=1}^{9} = \{0.24, 0.13, 0.011, 0.0008, 2.8\;10^{-5}, 8.8\;10^{-7}, 2.1\;10^{-8}, 4.1\;10^{-10}, 6.4\;10^{-12}\}. \qquad (20)$$

Из тождеств (18) – (20) можно сделать следующий вывод.

Вывод 1. *Коэффициент оптимальности решения по алгоритму 1 краевой задачи* (16) *в пространстве* $C_{[0,1]}$ *ограничен*

$$C_n(algorithm_1, task\ (16), C_{[0,1]}\,) = (1 + o(1))\,(5 + \alpha_{2[n/2]}), \qquad (21)$$

$$\{\alpha_{2i}\}_{i=1}^{9} = \{0.3, 1.3, 0.9, 0.4, 1.1, 0, -0.1, -0.2, -0.4\}.$$

Норма погрешности решения краевой задачи (16) по алгоритму 1 в пространстве $C^2_{[0,1]}$ убывает с ростом параметра n алгоритма

$$\| y - y_{2i+3} \|_{C\text{^}2_[0,1]} = \| y - y_{2i+2} \|_{C\text{^}2[0,1]} = \| y'' - y''_{2i+2} \|_{C[0,1]} = \| u - u_{2i} \|_{C[0,1]},$$
$$\{\| u - u_{2i} \|_{C[0,1]}\}_{i=0}^{8} = \{86, 22, 2.8, 0.24, 1.4\ 10^{-2},$$
$$5.8\ 10^{-4}, 1.8\ 10^{-5}, 4.5\ 10^{-7}, 8.8\ 10^{-9}\}. \qquad (22)$$

Отличные от нуля коэффициенты Фурье – Чебышева прозводных y' и $y'' = u = 100\ y$ функции y (17) на отрезке [0,1], с ростом параметра n, регулярно убывают (18). Следовательно, имеет место цепочка тождеств:

$$E_n(y, C^2_{[0,1]}) = E_{n-2}(y'', C_{[0,1]}) = (1 + o(1))\ |\ a_{2i}(y'', [0,1])\ | = (100 + o(1))\ |\ a_{2i}(y, [0,1])\ |$$

и из тождеств (18), (22) можно сделать следующий вывод.

Вывод 2. *Коэффициент оптимальности решения по алгоритму 1 краевой задачи* (16) *в пространстве* $C^2_{[0,1]}$ *монотонно убывает к единице (с ростом параметра n)*

$$C_n(algorithm_1, task\ (16), C^2_{[0,1]}) = (1 + o(1))\ (1 + \alpha_{2\ [n/2]}), \qquad (23)$$
$$\{\alpha_{2i}\}_{i=1}^{10} = \{0.8, 0.6, 0.3, 0.2, 0.1, 0.08, 0.06, 0.05, 0.04, 0.03\}.$$

Коэффициент оптимальности (6) сплайн-метода [4]. По методу [4, с. 63] вычисляют сеточную функцию

$$S_q = \{S_q(i/q)\}_{i=0}^{q} = \text{spline_method}(task\ (16), [0,1], q) \approx \{y(i/q)\}_{i=0}^{q}.$$

Рассмотрим метод решения краевой задачи (16) – интерполяция P_q многочленом порядка q сеточной функции S_q (вычисленной сплайн-методом [4])

$$P_q[S_q] = P_q[\text{spline_method}(task\ (16), [0,1], q)]. \qquad (24)$$

Норма погрешности этого метода в пространстве $C_{[a,b]}$, очевидно, не меньше сеточной нормы погрешности сплайн-метода [4]

$$\max_{x \in [0,1]} |\ y(x) - P_q[\ S_q]| \geq \max_{i=0,\dots,q} |\ y(\ i/q\ ) - S_q(\ i/q\ )\ |.$$

Поэтому значение коэффициента оптимальности метода $P_q[S_q]$ (24) на краевой задаче (16) не меньше

$$C_q(P_q[S_q], (16), C_{[0,1]}) \geq \max_{i=0,\dots,q} |\ y(i/q) - S_q(i/q)\ |\ /\ E_q(y, C_{[0,1]}).$$

Сравнение метода [4] и алгоритма 1. Погрешность аппроксимации точного решения (17) краевой задачи (16) сеточной функцией S_q принимает [4, с. 63] следующие значения:

$$\max_{i=0,\dots,10} |\ y(i/10) - S_{10}(i/10)\ | = 10^{-2}\ , \max_{i=0,\dots,20} |\ y(i/20) - S_{20}(i/20)\ | = 3\ 10^{-3}.$$

Поэтому из тождеств (18), (19) можно сделать следующий вывод.

Вывод 3. *Коэффициент оптимальности метода* (24) *на задаче* (16) *растет экспотенциально (с ростом числа узлов сетки)*

$$C_{10}(P_{10}[S_{10}], (16), C_{[0,1]}) \geq 2000,\ C_{20}(P_{20}[S_{20}], (16), C_{[0,1]}) \geq 10^{12}/6,$$
$$C_q(P_q[S_q], (16), C_{[0,1]}) = O(z^q),\ z > 2.$$

Такая же зависимость коэффициента оптимальности метода от параметра q (порядка вычисляемого многочлена или числа узлов сетки) имеет место и в случае решения краевой задачи (16) по другим методам с насыщением. На основании этой зависимости и тождеств (21), (23) можно сделать следующее заключение.

Вывод 4. *Решение краевой задачи* (16) *по алгоритму 1 оптимально для символьных преобразований в СКА.* (Это основное отличие алгоритма 1 от сплайн-метода [4, с. 63], метода (24) и других методов с насыщением.)

3 Эквивалентность алгоритма 1 и алгоритма [2]

Алгоритм [2]

1. Вычислить интегральный оператор $K[u]$ (4) и многочлен $y_{k\text{-}1}$ (3).
2. Преобразовать систему уравнений (1), (2) в линейное функциональное уравнение

$$L[u] + g = 0,\ \ L[u] = D[K[u]],\ \ g = D[y_{k\text{-}1}]+ G,\ \ u = y^{(k)}. \qquad (25)$$

3. Вычислить аппроксимацию уравнения (25) по a-методу [3] с параметром $p= n - k$

$$L[u_p \in P_p\ ] + g + (E_{m,p}(z(x)) \in H_{m\backslash p}) = 0, \qquad (26)$$

где дополнительный многочлен имеет параметр

$$m = \deg(L[\ u_p \in P_p] + g). \qquad (27)$$

4. Решить уравнение (26) и вычислить многочлены u_p, $E_{m,p}(z(x))$.
5. Вычислить преобразование $K[u]+y_{k\text{-}1}$ (3), (4) многочлена u_p

$$y_n = K[u_p] + y_{k\text{-}1} \qquad (28)$$

– искомую аппроксимацию решения y краевой задачи (1), (2).

Замечание 1. Согласно определению оператора K (4) и многочлена $y_{k\text{-}1}$ (3), многочлен y_n (28) является решением СЛАУ (8).

Теорема 1. *Пусть:*

– *оператор* $K[u]$ (4) *преобразует моном* x^i , $i \in N$ *в полином*

$$K[x^i] = b_i\, x^{i+k} + \ldots + e,\ \ b_i \neq 0, \qquad (29)$$

– *параметр алгоритма* 1 (*и алгоритма* [2])

$$n \geq 2\,k\ ,\ \ k = \text{ord_equ}(D[y]). \qquad (30)$$

Тогда уравнение (26) *тождественно замене в уравнении* (7) *многочлена* $y_n \in P_n$ *на многочлен* $K[u_p \in P_p\ ] + y_{k\text{-}1}$, *где* $p = n - k$ – *уравнению*

$$\begin{aligned}&\text{subs}((y_n \in P_n) = K[u_p \in P_p] + y_{k\text{-}1}\,, D[y_n \in P_n] + G + (E_{m,p}(z(x)) \in H_{m\backslash p}) = 0\ )\\ &\quad = (D[K[K[u_p \in P_p] + y_{k\text{-}1}] + G + (E_{m,p}(z(x)) \in H_{m\backslash p}),\end{aligned} \qquad (31)$$

где $m = \deg(D[y_n \in P_n] + G)$.

Доказательство. Согласно определению уравнения (25), первое слагаемое уравнения (7) тождественно первому слагаемому уравнения (26).

Параметр p дополнительного многочлена $E_{m,p}(z(x))$ уравнений (7), (31) и (26) тождественен по определению.

Тождество первого параметра m (10) и (27) дополнительных многочленов этих уравнений

$$\deg(D[y_n \in P_n] + G) = \deg(L[u_p \in P_p] + g),\ \ p = n - k$$

в случае $\deg(D[y_n \in P_n] + G) = \deg(G)$ очевидно.

Тождество первого параметра m (10) и (27) дополнительных многочленов этих уравнений в случае $\deg(D[y_n \in P_n] + G) = \deg(D[y_n \in P_n])$ непосредственно следует из очевидного тождества

$$\deg(D[K[u_p \in P_p] + y_{k\text{-}1}]) = \deg(D[K[u_p \in P_p]]),$$

неравенств

$$\deg(D[K[u_p \in P_p]]) \leq \deg(D[y_n \in P_n]),\ \ n = p + k, \qquad (32)$$

$$\deg(D[K[u_p \in P_p]]) \geq \max\{\deg(D[\ x^k\ ]),\ \ldots,\ \deg(D[\ x^n\ ])\}, \qquad (33)$$

тождества

$$\deg(D[y_n \in P_n]) = \max\{\deg(D[1]), \deg(D[x]), \ldots, \deg(D[x^n])\} \quad (34)$$

и тождества, имеющего место при выполнении условия теоремы $n > 2k$,

$$\deg(D[y_n \in P_n]) = \max\{\deg(D[x^k]), \ldots, \deg(D[x^n])\}. \quad (35)$$

Тождества (34), (35) доказаны нами в работе [5].

Лемма 1. *Пусть* $D[y]$ *– линейный дифференциальный оператор с многочленными коэффициентами вида* (1) *и для оператора* $K[u]$ (4) *справедливо тождество* (29).

Тогда справедливы неравенства (32), (33).

Доказательство. Согласно определению (4) оператора $K[u]$, многочлен $K[u_p \in P_p]$ имеет порядок

$$\deg(K[u_p \in P_p]) = p+k = n.$$

Поэтому он является частным случаем многочлена $y_n \in P_n$. Следовательно, многочлен $D[K[u_p \in P_p]]$ является частным случаем многочлена $D[y_n \in P_n]$ и порядок многочлена $D[K[u_p \in P_p]]$ не превышает порядок многочлена $D[y_n \in P_n]$ – имеет место неравенство (32).

Согласно тождеству (29), имеет место тождество

$$K[c_0] = c_0\,K[1] = c_0\,(b_0\,x^k + \ldots + e_0),\ \ b_0 \neq 0$$

и неравенство (33) в случае $p = 0$ очевидно

$$\deg(D[K[c_0]]) = \deg(D[K[1]]) = \max\{\deg(D[x^k]), \ldots\} \geq \deg(D[x^k]).$$

Операторы $D[y]$ и $K[u]$ – линейные. Поэтому имеет место тождество

$$D[K[u_s \in P_s]] = D[K[u_{s-1} \in P_{s-1}]] + c_s\,D[K[x^s]].$$

Согласно этому тождеству и правилам вычисления порядка многочлена, имеет место тождество

$$\deg(D[K[u_s \in P_s]]) = \max\{\deg(D[K[u_{s-1} \in P_{s-1}]], \deg(D[K[x^s]])\}. \quad (36)$$

Согласно тождеству (29), имеет место неравенство

$$\deg(D[K[x^s]]) \geq \deg(D[x^{k+s}]).$$

Если неравенство (33) имеет место для $p = s - 1$, то из этого неравенства и тождества (36) следует справедливость неравенства (33) в случае $p = s$.

4 Сходимость алгоритма 1 и оценки погрешности

Согласно теореме 1, для алгоритма 1 имеют место результаты [2] исследования алгоритма [2]. Следовательно, для достаточно широкого класса ЛДУМК (1) и краевых условий (2), эти результаты:

– доказывают существование решения y_n (9) системы аппроксимирующих уравнений (7), (8) алгоритма 1;

– доказывают сходимость последовательности многочленов y_k, y_{k+1},… (9) к точному решению y краевой задачи для ЛДУМК (1), (2);

– устанавливают точную и конструктивную апостериорную оценку нормы погрешности алгоритма 1 в пространстве $C_{[a,b]}$ и $C^k_{[a,b]}$;

– устанавливают точную и конструктивную априорную оценку коэффициента оптимальности алгоритма 1 в пространстве $C^k_{[a,b]}$;

– доказывают ограниченность коэффициента оптимальности алгоритма 1 в пространстве $C^k_{[a,b]}$.

5 Вычисление оценок оптимальности алгоритма 1

Согласно теореме 1, алгоритмы [2] вычисляют:
– точную оценку нормы погрешности алгоритма 1 в пространстве $C_{[a,b]}$ и $C^k_{[a,b]}$;
– точную оценку коэффициента оптимальности алгоритма 1 в пространстве $C^k_{[a,b]}$ для достаточно широкого класса ЛДУМК (1) и краевых условий (2).

6 Программирование алгоритма 1 в СКА

Операторы СКА не преобразуют систему уравнений (1), (2) в систему уравнений (7), (8) и не решают систему уравнений (7), (8). Поэтому мы построим алгоритм 2 преобразования краевой задачи для ЛДУМК (1), (2) в искомый многочлен y_n , и докажем эквивалентность аппроксимации по этому алгоритму краевой задачи для ЛДУМК (1), (2) системе уравнений (7), (8).

Алгоритм 2

Вход: $D[y] + G = 0$ (1), $Cond(y)$ (2), $[a, b]$, n.
Выход: Многочлен y_n (9).
Преобразования:
1. Вычислить оператор $D[y] + G$ – левую часть ЛДУМК (1).
2. Вычислить k – порядок старшей производной оператора $D[y]$.
3. Вычислить многочлен $y_n \in P_n$.
4. Вычислить многочлен $D[y_n \in P_n] + G$.
5. Вычислить порядок m (10) многочлена $D[y_n \in P_n] + G$.
6. Вычислить параметр p (11) дополнительного многочлена.
7. Вычислить многочлен $E_{m,p}$ (14) (с символьными коэффициентами).
8. Вычислить многочлен $z(x)$ (13).
9. Вычислить многочлен $E_{m,p}(z(x)) \in H_{m\backslash p}$ (12).
10. Вычислить аппроксимацию оператора $D[y] + G$ (1) – многочлен

$$D[y_n \in P_n] + G + (E_{m,p}(z(x)) \in H_{m\backslash p}). \quad (37)$$

11. Вычислить систему линейных алгебраических уравнений

$$S = \{t_i\,(D[y_n \in P_n] + G + (E_{m,p}(z(x)) \in H_{m\backslash p})) = 0,\ \ i = 0, \ldots, m\}, \quad (38)$$

где $t_i(P)$ – коэффициент монома порядка i полинома P.
12. Вычислить объединение СЛАУ (38) и СЛАУ (8). Для $i = 1, \ldots, k$ вычислить:
 12.1. Условие (2) с номером i.
 12.2. Линейное дифференциальное уравнение этого условия

$$D_i[y] + G_i = 0. \quad (39)$$

 12.3. Левую часть уравнения (39) – оператор $D_i[y] + G_i$.
 12.4. Многочлен $D_i[y_n \in P_n] + G_i$.
 12.5. Точку d_i задания краевого условия i (2).
 12.6. Левую часть уравнения i СЛАУ (8) – $(D_i[y_n \in P_n] + G_i)\,|_{x=d_i}$.
 12.7. Уравнение i СЛАУ (8)

$$S_i = ((D_i[y_n \in P_n] + G_i)\,|_{x=d_i} = 0). \quad (40)$$

 12.8. Объединение уравнения (40) с объединением СЛАУ (38) и вычисленной ранее частью СЛАУ (8)

$$\mathrm{conc}(S_i = 0,\ \{S\ (38),\ S_1 = 0, \ldots, S_{i-1} = 0\}).$$

13. Решить объединение СЛАУ (8) и СЛАУ (38) – вычислить значения коэффициентов $c_0, \ldots, c_n$, $\tau_1, \ldots, \tau_{m-p}$ многочленов y_n, $E_{m,p}(z(x))$ (9):

$$Coef = solve(S\,(38), (8)) = \{coef(y_n), coef(E_{m,p})\}. \qquad (41)$$

14. Вычислить многочлен y_n (5) с числовыми коэффициентами – преобразовать значения коэффициентов этого многочлена $coef(y_n) = \{d, \ldots, e\}$ (41) в сумму мономов:

$$y_n = ser(coef(y_n)) = d + \ldots + e\,x^n. \qquad (42)$$

Доказательство. Операторы $D_i[y]$ условий (2) – линейные. Поэтому левая часть уравнения i системы (8) имеет вид

$$(D_i[y_n \in P_n]) + G_i)_{x = d_i} = A_i\,(y_n \in P_n)^{(k_i)}{}_{x = d_i} + \ldots + C_i\,(y_n \in P_n\,)_{x = d_i} + G_i\,,$$

и это уравнение (относительно коэффициентов многочлена $y_n \in P_n$) – линейное.

Функционал t_i – линейный и оператор $D[y]$ – линейный. Поэтому левая часть уравнения $i = 0, 1, \ldots, m$ системы (38) имеет вид

$$t_i(D[y_n \in P_n]) + t_i(G) + t_i(E_{m,p}(z(x)) \in H_{m\backslash p}),$$

где слагаемые являются линейными формами коэффициентов многочленов $y_n \in P_n$ и $E_{m,p}(z(x)) \in H_{m\backslash p}$:

$$t_i(D[y_n \in P_n]) = t_i(D[1])\,c_0 + \ldots + t_i(\,D[x^n])\,c_n\,,\ t_i(G) = Const,$$
$$t_i(E_{m,p}(z(x)) \in H_{m\backslash p}) = t_i(\mathrm{cheb}(p{+}1, z(x))\,\tau_1 + \ldots + t_i(\mathrm{cheb}(m, z(x))\,\tau_{m-p}.$$

Многочлен (37) тождественен левой части уравнения (7) по определению. Этот многочлен имеет порядок m (10). Следовательно, уравнение (7) эквивалентно СЛАУ (38) и многочлен y_n (42) тождественен решению y_n системы уравнений (7), (8).

7 Программирование алгоритма 2 в системе APS

Структура данных на входе

1. ЛДУМК (1) имеет вид

```
A * dif(y, k) + ... + C * y + G = 0,
```

где y – атом, коэффициенты `A ,…, C, G` являются многочленами переменной (атома) x (термами) и имеют вид, естественный для математики.

2. Краевые условия (2) являются списком вида

```
...,(x = d_i, A_i * dif(y, k_i) +...+ C_i * y + G_i = 0),...
```

3. Отрезок аппроксимации $[a\,, b]$ определяет список `(a , b)`.

4. Параметр n алгоритма является целым числом.

Структура данных на выходе

Многочлен y_n (42) имеет числовые коэффициенты и вид, естественный для математики

```
d + ... + f * x ^ n.
```

APLAN-процедура. Алгебраическая спецификация алгоритма 2.

```
let(LDUMK, Dy = 0);   /* Dy = D[y] + G */
k := ord_equ(Dy);    /* порядок Dy */
y_n := main_pol(n);    /* y_n \in P_n */
Dn := canplf(sub_du(Dy, y_n)); /* D[y_n] + G */
m := deg(canplf(ein_pol(Dn))); /* deg(D[y_n]+G) */
p := n - k;
```

```
Em := Enl(n,m-n+k,k);      /* E_{m,n} */
b := arg(interval, 2); a := arg(interval, 1);
h := b - a;
z := canplf((2/h) * (x + (-1) * a) + -1); /* z(x) */
Em --> canplf(subs(x = z, Em )); /* E_{m,n}(z(x))*/
Dn --> canplf(Dn + Em);  /* D[y_n] + G + E_m */
S := pol_equ(Dn, m);  /* СЛАУ - аппр. ЛДУМК */
 /* АППРОКСИМАЦИЯ условий (2) */
for(i := 1, i < k, i := i+1,
 Cond_i --> arg(Cond, 1);
 Cond --> arg(Cond, 2);
 point --> arg(Cond_i, 1);     /* x = d_i */
 LDUMK_i --> arg(Cond_i, 2); /* D_i[y] + G_i = 0 */
 let(LDUMK_i, Dy = 0);  /* Dy = D_i[y] + Gi*/
 Dn --> canplf(sub_du(Dy, y_n)); /* D_i[y_n] + Gi*/
 S_i --> subs(point, Dn); /* D_i[y_n]|_{x=d_i} + Gi*/
 S --> conc(S, copy(S_i) = 0)
 );
point --> arg(Cond, 1);      /* x = d_k */
LDUMK_i --> arg(Cond, 2);     /* D_k[y] = 0 */
let(LDUMK_i, Dy = 0);  /* Dy = D_k[y] + G_k */
Dn --> canplf(sub_du(Dy , y_n)); /* D_k[y_n] +G_k */
S_i --> subs(point,Dn); /* D_k[y_n])|_{x=d_k} + Gk */
S --> conc(S, copy(S_i) = 0); /* СЛАУ (38), (8) */
Xn := c; Coef := solve(S);  /* решение СЛАУ */
y_n := ser(n, Coef);   /* аппроксимация y */
```

Структура выхода операторов APLAN-процедуры

Многочлен $y_n \in P_n$ APLAN-процедура вычисляет в виде

`c(0) + c(1) * x +...+ c(n) * x ^ n,`

где `x` и коэффициенты `c(0),..., c(n)` являются атомами. Оператор $D[y]$ (1) преобразует этот многочлен в многочлен $D[y_n \in P_n]$ порядка m (10). Оператор `canplf` APLAN-процедуры приводит этот многочлен к сумме слагаемых вида

`c(i) * x ^ j $ b`, $i = 0,\ldots,n, j = 0,\ldots, m,$ (43)

где $ – операция умножения терма на число.

Дополнительный многочлен $E_{m,p}$ (14) APLAN-процедура вычисляет в виде

`c(n + 1) * cheb(p + 1,x) +...+ c(m + k) * cheb(m,x),`

где коэффициенты `c(n + 1),..., c(m + k)` являются атомами, и многочлены Чебышева первого рода – `cheb(i,x)` – вычисляет внутренний оператор солвера *gr_solve.exe* системы APS. Этот многочлен и его преобразование (12) имеют порядок m (10). Оператор `canplf` APLAN-процедуры приводит многочлен $E_{m,n}(z(x)) \in H_{m\backslash n}$ (12) к сумме слагаемых вида

`c(i) * x ^ j $ b`, $i = n+1, \ldots, m+k$, $j = 0,\ldots, m.$ (44)

Поэтому объединение СЛАУ (8) и СЛАУ (38) является списком уравнений вида

```
c(m + k) $ f +...+ c(0) $ e + d = 0,
```

и решение (41) объединения СЛАУ (8) и СЛАУ (38) является списком тождеств

```
c(0) = h ,..., c(n) = p ,..., c(m + k) = q.
```

8 Исследование APLAN-процедуры

Теорема 2. *Пусть коэффициенты многочленов и другие параметры исходных данных APLAN-процедуры являются целыми или рациональными числами.*

Тогда APLAN-процедура вычисляет многочлен y_n *(42) точно.*

Доказательство. APLAN-процедура имеет известные операторы. Эти операторы в системе APS преобразуют рациональные числа (коэффициенты многочленов) точно – не вносят в результат преобразований погрешности от выполнения арифметических операций компьютером с фиксированной длиной машинного числа.

Теорема 3. *APLAN-процедура имеет по параметру n полиномиальную сложность*

$$(m + 1)\ Q(canplf, m) + O(n^3),\ \ m = \deg(D[y_n \in P_n] + G\ ) = n + O(1), \qquad (45)$$

где $Q(\ canplf,\ m\ )$ *– сложность преобразования оператором canplf многочлена P: многочлен* $canplf(P)$ *является суммой мономов вида* (43) *и* (44) .

Доказательство. APLAN-процедура линейная. Следовательно, вычислительная сложность этой процедуры тождественна сумме вычислительной сложности ее операторов. Операторы APLAN-процедуры, за исключением оператора canplf, имеют по параметру n полиномиальную сложность $O(n^s)$, $s \le 3$. Наибольшую вычислительную сложность имеет оператор pol_equ. Вычислительная сложность оператора pol_equ тождественна оценке (45). Оператор canplf преобразует многочлены $D[y_n \in P_n] + G$, $D_i[y_n \in P_n] + G_i$, (12) и (37) в сумму мономов вида (43) и (44).

9 Вычислительный эксперимент с APLAN-процедурой

Описание краевой задачи (16) для APLAN-процедуры

```
process[1] := (
LDUMK := (dif(y, 2) + (-100) * y = 0);
Cond := ((x = 0, y + -1 = 0), (x = 1, y + -1 = 0));
    interval := (0, 1);  ...);
```

Результат преобразования процедурой задачи (16) и параметра $\boldsymbol{n = 2}$

```
y_n := rat(200,29) * x ^ 2 + 1 + rat(-200, 29) * x.
```

Заключение

Построенный в работе алгоритм τ-метода Ланцоша оптимально решает в СКА достаточно широкий класс многоточечных линейных краевых задач для линейных дифференциальных уравнений порядка k с многочленными коэффициентами (1), (2).

Дополнение

1. **Алгоритм 2'.** Модификация алгоритма 2 – дополнительный многочлен СЛАУ (38) имеет ортогональный базис пространства Гильберта H .

2. **Модификация APLAN-процедуры.** Оператор вычисления многочлена Чебышева заменен оператором вычисления многочлена базиса алгоритма 2' на отрезке [–1,1]. Эта модификация APLAN-процедуры реализует алгоритм 2' в системе APS.

3. Для алгоритма 2' справедлив аналог теоремы 1. В этом аналоге уравнение (26) имеет дополнительный многочлен с базисом пространства *H*.

Литература

1. Ланцош К. Практические методы прикладного анализа. – М.: Физматгиз, 1957. – 584 с.
2. Денисенко П.Н. Оптимальный метод решения краевых задач в системах компьютерной алгебры и его программирование в системе APS // Искусственный интеллект. – 2006. – № 4. – С. 154-173.
3. Дзядык В.К. Аппроксимационные методы решения дифференциальных и интегральных уравнений. – Киев: Наукова думка, 1988. – 387 с.
4. Алберг Дж., Нильсон Э., Уолш Дж. Теория сплайнов и ее приложения. – М.: Мир, 1972. – 316 с.
5. Денисенко П.Н. Реализация тау-метода Ланцоша в APS // Искусственный интеллект. – 2005. – № 1. – С. 48-58.

П.М. Денисенко

Алгоритм розв'язання крайових задач в системах комп'ютерної алгебри за τ-методом Ланцоша

У статті побудовано алгоритм τ-методу Ланцоша для розв'язання багатоточкових лінійних крайових задач для лінійних диференціальних рівнянь порядку *k* з багаточленними коефіцієнтами. За ним в комп'ютерних системах символьного перетворення обчислюють багаточлен порядку *n*. Доведена еквівалентність цього алгоритму та алгоритму застосування а-методу Дзядика. Результати теорії а-методу доводять існування розв'язання вихідної задачі за цим алгоритмом, збіжність послідовності таких розв'язань (з ростом параметру *n* алгоритму) до точного розв'язання крайової задачі, точні і конструктивні апріорні та апостеріорні оцінки похибки в просторах $C_{[a,b]}$ і $C^k_{[a,b]}$ для досить широкого класу крайових задач.

P.N. Denisenko

The Algorithm of Solving the Boundary-value Problems in the Computer Algebra Systems Using the Lanczos τ-method

The Lanczos τ-method algorithm of solving the multipoint boundary-value problem for linear differential equations of order k with polynomial coefficients is developed in the article. The polynomial of order *n* is computed by this algorithm in the computer algebra systems. We proved the equivalence of this algorithm to the V. K. Dzyadyk a-method algorithm. The research results of the a-method prove the solution existence for the initial problem by the algorithm, the convergence of these solutions sequence (with the increase of the algorithm parameter *n*) to the exact boundary-value problem solution, the exact and constitutive estimates a priori and a posteriori in the spaces $C_{[a,b]}$, $C^k_{[a,b]}$.

Статья поступила в редакцию 25.07.2007.